Remarks

This REQUEST FOR CONTINUED EXAMINATION and REPLY is in response to the Final Office Action mailed February 1, 2007, and the Advisory Action mailed July 20, 2007. A Request for Extension of Time is also included herewith, together with the appropriate fee. No fee is due for the addition of new claims.

Applicant thanks the Examiner for the detailed comments provided by the Examiner in the Advisory Action mailed July 20, 2007.

I. Summary of Examiner's Rejections

Prior to the Office Action mailed February 1, 2007, Claims 1, 4, 6-8, 10, 11, 14, 16-18, 20, 21, 24, 26-28 and 30-33 were pending in the Application. In the Office Action, the Specification and Claims 1, 11 and 21 were objected to for various informalities. Claims 1, 4, 6-8, 10, 11, 14, 16, 18, 20, 21, 24, 26, 28 and 30 were rejected under 35 U.S.C. 103(a) as being unpatentable over Berg et al. (U.S. Publication No. 2004/0088681, hereinafter Berg) in view of Rich et al. (U.S. Publication No. 2002/0178439, hereinafter Rich). Claims 7, 17, 27 and 31-33 were rejected under 35 U.S.C. 103(a) as being unpatentable over Berg in view of Rich, and further in view of McIntyre (U.S. Patent No. 6,178,546).

II. Summary of Applicant's Amendments

The present Response amends the Specification, and Claims 1, 11, 21 and 31-33; and adds new Claims 34-42, leaving for the Examiner's present consideration Claims 1, 4, 6-8, 10, 11, 14, 16-18, 20, 21, 24, 26-28 and 30-42.

III. Objections to the Specification

In the Office Action mailed February 1, 2007, the Specification was objected to for various informalities. Accordingly, the Specification has been amended as shown above. Applicant respectfully submits that the proposed amendments correct informalities in the Specification and that no new matter is being added.

IV. Objections to the Claims

In the Office Action mailed February 1, 2007, Claims 1, 11 and 21 were objected to for various informalities. Accordingly, the claims have been amended as shown above to correct the informalities. Reconsideration thereof is respectfully requested.

V. Claim Rejections under 35 U.S.C. §103(a)

In the Office Action mailed February 1, 2007, Claims 1, 4, 6-8, 10, 11, 14, 16, 18, 20, 21, 24, 26, 28 and 30 were rejected under 35 U.S.C. 103(a) as being unpatentable over Berg (U.S. Publication No. 2004/0088681) in view of Rich (U.S. Publication No. 2002/0178439). Claims 7, 17, 27 and 31-33 were rejected under 35 U.S.C. 103(a) as being unpatentable over Berg in view of Rich, and further in view of McIntyre (U.S. Patent No. 6,178,546).

Claim 1

Claim 1 has been amended to more clearly define the embodiment therein. As amended, Claim 1 defines:

> *1. (Currently Amended) A system for organization of software application files during development and subsequent deployment of the software application to a server, comprising:*
>
> *a split directory structure stored on a computer medium that stores files for a software application, wherein, for each software application, the split directory structure includes both a source folder that stores editable source files as part of the software application, and a corresponding output folder that stores compiled files as part of the software application, and wherein the split directory is accessed as a virtual file that provides an abstraction over the two folders therein;*
>
> *a server upon which the software application will be deployed; and*
>
> *a deployment tool that allows a user to specify the output folder during deployment of the software application, wherein during the deployment the server interprets the software application as a union of both the source folder and the output folder, recognizes the split directory structure, and deploys the application by making requests to the virtual file which checks first the source folder and then the corresponding output folder for software application files, before deploying the software application files to the server.*

As presently amended, Claim 1 defines a system for organization of software application files during development and subsequent deployment of the software application to a server. The system comprises a split directory structure that stores files for a software application. For each software application, the split directory structure includes both a source folder that stores editable source files, and a corresponding output folder that stores compiled files, wherein the split directory is accessed as a virtual file that provides an abstraction over the two folders therein. During deployment, the server interprets the software application as a union of both the source folder and the output folder, recognizes the split directory structure, and deploys the application by making requests to the virtual file, which checks first the source folder and then the corresponding output folder for software application files, before deploying the software application files to the server.

The advantages of the embodiment defined by Claim 1 include that, during deployment of the software application, the software application is interpreted as a union of both the source folder and the output folder. This approach requires no copying, in that the server can read source files (for example JSP's, XML descriptors, html images, etc.) directly from the split directory structure, without having to first copy them to a build directory. Similarly, the server receiving the build can see both the /build folder, and the /source folder. This allows, for example, Web files to be changed and redeployed in place within the source folder, without having to rebuild the entire software application. It also allows, for example, the output folder to be deleted to remove the latest build of the software application, and then recreated to create a new build.

Berg discloses a method and system for dynamically mapping archive files in an enterprise application. As disclosed therein, the highest-level project, (the project that "contains" the nested archives, or contains references to other projects that conceptually represent nested archives) is referred to generically as a "container project". Two examples of container projects in a J2EE environment are EAR projects, that represent enterprise applications or EAR files, and Web projects, that represent web applications or WAR files. The application server, however, expects the files to be in a hierarchical directory structure as noted above. (Paragraph [0010]). In accordance with the [present] invention, mapping information that describes the mapping of referenced projects to their container project is included in the container project, using a "module mapping" file. (Paragraph [0019]).

Rich discloses a method and system for providing a programming interface for loading and saving archives in enterprise applications. As disclosed therein, the two environments, assembly and "runtime" (the running application server), are therefore dealing with two different physical file structures: assembly, with JARs; and runtime, with a mixture of expanded JARs in a directory tree structure and JAR files themselves. (Paragraph [0014]). At step 102, the user (running program) requests the loading of an archive having a given file path/name. At step 104, a determination is made if the requested file is an archive file type (e.g., .jar, .zip, .war, etc.) or a directory tree file. At step 106, a loading strategy for loading and displaying the file requested by the user ("open archive" command) is created based on the status of the file requested as having an archive structure or a directory tree. At step 108, a "virtual" archive to return to the calling method is created based upon the loading strategy. (Paragraphs [0025-0028]).

In the Office Action mailed February 1, 2007, (and the Advisory Action mailed July 20, 2007), it was submitted that Berg represents a split directory structure; and that the container project is developed in an IDE and includes a source folder that stores editable source files. It was further submitted that the files are considered editable source files at least because as they are stored in the IDE, ... the programmer can debug and modify [them]. It was also submitted that the virtual archive of Rich is analogous to the virtual file defined in Claim 1, in that it provides an abstraction over a directory structure.

However, Applicant respectfully submits that, while Berg appears to disclose that original files are stored in an IDE, there does not appear to be any description in Berg as to how, once files are generated or compiled, the generated or compiled files are accessed together with their corresponding source files. Instead, Berg appears to disclose that a higher-level or container project can be mapped to other, referenced projects. Applicant respectfully submits that neither Berg nor Rich, when considered alone or in combination, appear to disclose or suggest that, for a particular software application, both the source and the compiled files are stored in folders within a split directory that is accessed as a virtual file, and that the virtual file provides an abstraction over the two folders for that software application. Similarly, Applicant respectfully submits that neither Berg nor Rich appear to disclose or suggest that, during deployment, the server interprets the software application as a union of both the source folder and the output folder, recognizes the split directory structure, and deploys the application by making requests to the virtual file, which checks first the source folder, and then the corresponding output folder, for software application files, before deploying the software application files to the server. Claim 1 has been amended to more clearly define these features.

In view of the above comments, Applicant respectfully submits that Claim 1, as amended, is neither anticipated by, nor obvious in view of the cited references, when considered alone or in combination. Reconsideration thereof is respectfully requested.

Claims 11 and 21

The comments provided above with respect to Claim 1 are hereby incorporated by reference. Claims 11 and 21 have been similarly amended to more clearly define the embodiments therein. For similar reasons as provided above with respect to Claim 1, Applicant respectfully submits that Claims 11 and 21, as amended, are likewise neither anticipated by, nor obvious in view of the cited references, and reconsideration thereof is respectfully requested.

Claims 4, 6-8, 10, 14, 16-18, 20, 24, 26-28 and 30-33

Claims 4, 6-8, 10, 14, 16-18, 20, 24, 26-28 and 30-33 are not addressed separately, but it is respectfully submitted that these claims are allowable as depending from an allowable independent claim, and further in view of the amendments to the claims and the comments provided above. Reconsideration thereof is respectfully requested.

VI. Additional Amendments

Claims 34-42 have been newly added by the present Response. Applicant respectfully requests that new Claims 34-42 be included in the Application, and considered therewith.

VII. Conclusion

In view of the above amendments and remarks, it is respectfully submitted that all of the claims now pending in the subject patent application should be allowable, and reconsideration thereof is respectfully requested. The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

Enclosed is a PETITION FOR EXTENSION OF TIME UNDER 37 C.F.R. §1.136 for extending the time to respond up to and including August 1, 2007.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this reply, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: August 1, 2007

By: /Karl F. Kenna/
Karl F. Kenna
Reg. No. 45,445

Customer No.: 23910
FLIESLER MEYER LLP
650 California Street, 14th Floor
San Francisco, California 94108
Telephone: (415) 362-3800